Beam Physics Note
TRI-BN-24-20
August 3, 2024

Centering Beam into DRAGON Using the Wobbler

Omar Hassan, Emma Ghelfi, Alexander Katrusiak

TRIUMF

Abstract: Beam centeredness at DRAGON has been previously addressed by observing the centroid shifts on a parallel wire position intensity monitor when turning the quadrupoles on/off to include transverse kicks proportional to beam misalignment. We devised a new method which takes advantage of the high frequency readback of the harp, inducing a time-dependent current variation on the quads leading to a periodic oscillation of the centroids if the beam is not centered. Steering corrections into DRAGON then aim to damp the amplitude of this transverse beam oscillation, producing a useful tool for operators. This allows for the formal treatment of multi-objective optimization for BOIS, where the second objective is minimizing the centroid shifts.

1 Introduction

So far, Bayesian Optimization for Ion Steering (BOIS) has been focused on solving a single optimization problem: the current transmission. It is important to also consider other beam properties to ensure good beam quality. One such property is the beam centeredness, currently indirectly addressed using scaleBOIS [1], the plan moving forward is to use a formal multi-objective treatment to optimize both of these beam properties.

To address beam centeredness, we use the first DRAGON quadrupole doublet (Q1 and Q2 in figure 2) and a parallel wire position-intensity monitor: the DRAGON harp. The procedure is as follows:

The Wobbler

1. Induce a time dependent current variation in the quadrupoles.
2. Adjust beam centroid using upstream steerers.
3. Observe corresponding effect on the harp.
4. Repeat until quadrupole variation has no effect on the beam centroids.

The beam is locally centered once the quads no longer have an effect on the beam centroids. Implicit in this is the assumption that the beam does not deviate further past the quadrupoles. While it is possible to define the center on the harp and tune based on that, this method wouldn't account for a beam coming in at an angle. The quadrupole doublet ensures that the beam is centered.

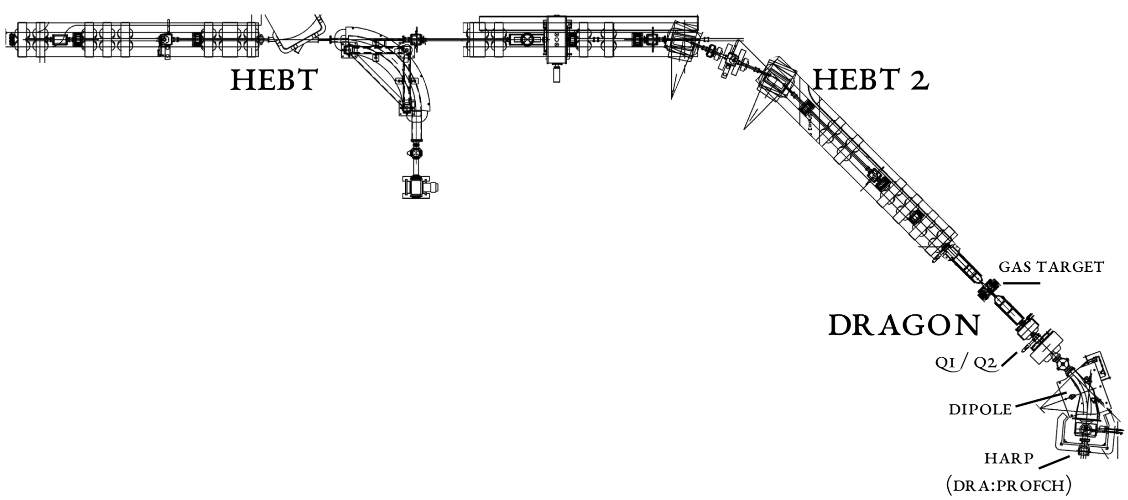

Figure 1: The beam line from the exit of the DTL to the DRAGON harp.

2 Setup

2.1 DRAGON Harp Readout

After discussion with DRAGON experimenters, it was determined that the harp readout is not known by DRAGON, i.e. the x and y limits in the array's readback in the control system. By moving the beam to the edges in x and y, we were able to determine the minimum and maximum values for the harp. The readout of `DRA:PROFCH:HARPARR` from jaya [**?**] is a 1D array of size 97, the first element (zeroth index) is unknown, the x range was found to be indices 1 to 48, while the y range was found to be indices 65 to 96. These ranges were determined by steering the beam away in each direction until the profile almost disappeared from the monitor (only showing on the last wire).

Table 1: `DRA:PROFCH` harp readout values.

Index Range	Length	Value(s)
0	1	Unknown.
1:48	48	X distribution.
49:64	16	N/A.
65:96	32	Y distribution.

Figure 2: Harp readout using the strip tools on EPICS.

2.2 Current Variation Techniques

Beam centeredness was previously determined by turning the quads on/off and observing if the centroids shift [3]. We require a different technique which makes use of the continuous feedback from the harp, this will both allow for easier manual centering and continuous data for a second objective in BOIS.

Three methods have been devised where the current applied to the quads is varied by a Δ value across time, with a maximum amplitude of 25% the set value (I_{set}). Alexander also determined that the quads have a ramping speed of 50 A/s, any current variation must then account for this.

2.2.1 Set Combinations

This method allows each quad to take one of three values: $\Delta = \pm 0.25 \cdot I_{\text{set}}$ or 0. Then we cycle through every combination between the two quads.

The one unique feature of this is that you would get every combination, however this is not much use to us. A significant disadvantage is that this method will likely exceed the ramp speed of the quads.

2.2.2 Sine Waves

The Δ can also be given by a sine wave

$$\Delta = \pm 0.25 \cdot I_{\text{set}} \cdot \sin \omega_0 t, \tag{1}$$

where the amplitude is 25% of the set current value, and the frequency ω can be found using:

$$\omega_0 \leq \frac{200}{I_{\text{set}}} \text{A/s}. \tag{2}$$

This is the sine function frequency, the total frequency will differ as there is a wait time added (t_w) due to the rate EPICS can receive instructions, and also due to recommendations from the controls group (K. Ezawa, private conversation). The total frequency is then:

$$\omega \leq \frac{1}{T_0 + t_w \cdot \frac{2\pi}{\delta}}, \tag{3}$$

where T_0 is the base sine function period, and δ is the step size. The sine wave method is convenient as it allows for fine tuning the current variations with smaller step sizes than the previous method. The primary downside of this method is the non constant rate of change: the current variation at the peaks is less significant than the center, which may mislead the operators or BOIS when centering the beam.

2.2.3 Triangle Waves

The triangle wave method is ideal as it has a constant rate of change while also allowing for smaller step sizes for the current variation. To be specific, we are using a symmetric triangle wave that is centered at 0:

$$\Delta = 0.25 \cdot I_{\text{set}} \cdot \left(1 - 2 \cdot \left| \frac{(t + \frac{T}{4}) \mod T}{\frac{T}{2}} - 1 \right| \right), \tag{4}$$

where T is the period of the wave.

2.2.4 Observed Beat Frequency

Given that the frequencies of the current oscillations are calculated from the set current values, they are in principle (and by definition) never equal. This means the path of the (x,y) point given from the harp readout will repeat at the beat frequency of these two oscillations. This is useful to track because it gives a period for which an operator should observe over to ensure the centroid's motion has truly been restricted.

Using eqn.3, and defining $T_w = t_w \frac{2\pi}{\delta}$, this total period for Q1 and Q2 together is:

$$T_{beat} = \left| \frac{(T_1 + T_w)(T_2 + T_w)}{T_1 - T_2} \right|, \tag{5}$$

where T_1 and T_2 are the base sine wave periods for Q1 and Q2, respectively.

2.3 Web Interface

The final part of the setup involves the web interface. Figure 3 shows an elementary version of the interface, where the primary focus points are on the quads and the harp centroids.

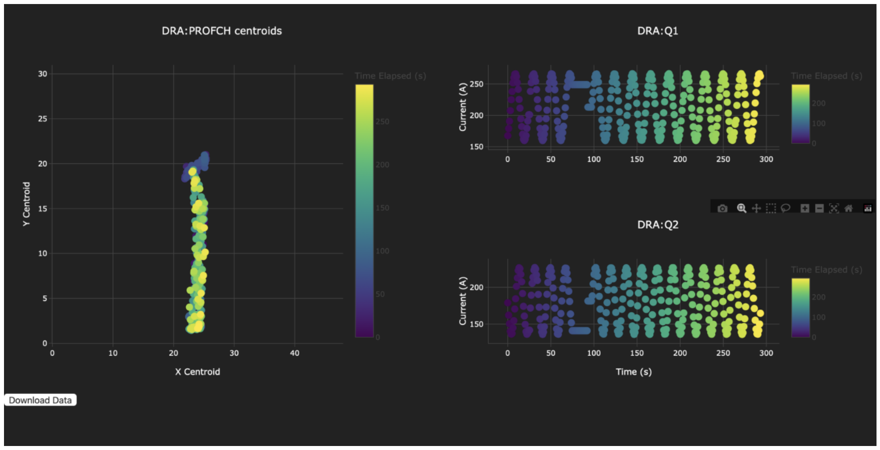

Figure 3: Web interface for the wobbler. On the left is a plot of the x and y centroids obtained from the harp distributions over time. The right hand side shows how the quads vary with time.

Of particular note, figure 3 shows a beam well aligned in x while significantly misaligned in y. The plan is to eventually add this as an extension to MCAT [4]. Following operator feedback, this will be a much more convenient tool to center beam into DRAGON.

3 Results

This tool is intended to be used by operators to manually center the beam, and by BOIS to address beam centeredness directly.

3.1 Manual Centering

RIB Operators tested the interface to center beam into DRAGON during a July 2024 machine development shift. The main difference from the typical operational process is that the sine wave quadrupole current was used, instead of manually cycling the quadrupoles on and off. Figure 4 shows the end result after centering over the last 25 points, where the centroid variation in both x and y was minimized.

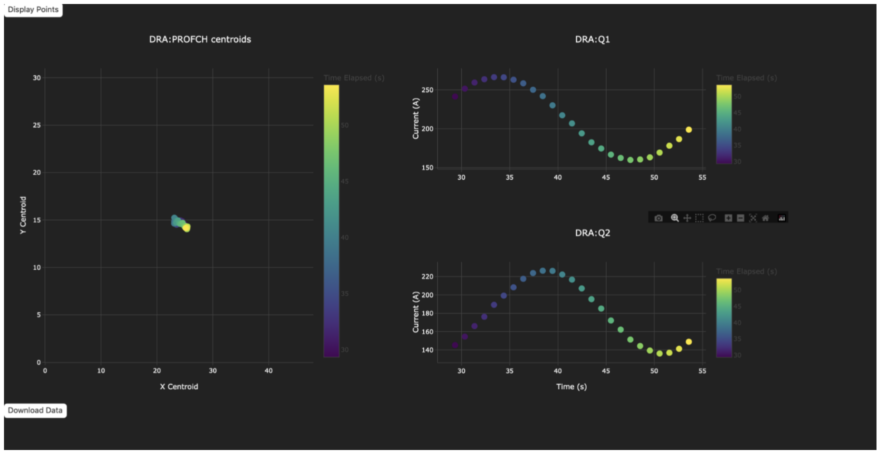

Figure 4: Web interface for the wobbler. On the left is a plot of the x and y centroids obtained from the harp distributions over time. The right hand side shows how the quads vary with time.

Operator feedback was very positive, seeing this method as a vast improvement from simply turning the quads on/off to check for beam centeredness. A consistent piece of feedback that we received from multiple operators was regarding the web interface, the operators stated that an indicator showing the starting transmission and the current transmission would be useful.

3.2 Equipment Issues

A notable issue that persisted with Q1 was that the setpoint value did not match the readback, the readback was consistently 9% higher. This issue doesn't matter so much for our purposes as it's a linear issue.

The other primary issue also involved Q1 where it appeared to trip off the power supplies randomly. We attempted to diagnose the problem to no avail, both our program and manual stress testing did not consistently trip off Q1, and of particular note, Q2 never tripped off once.

These issues were reported to the power supply group, an e-fault was submitted (#17200), this seemingly addressed the second issue but the 9% difference still remains.

4 Future Work

This beam note can be concluded by focusing on a few topics we wish to address in the future:

- Update the web interface to include suggested improvements by the operators. Implement as part of MCAT.
- Further test the wobbler focusing on a triangle wave, with variable Δ amplitudes based on the set current I_{set}.
- Implement the ability to optimize on different objectives in BOIS, where the second objective will be minimizing the standard deviation of the last 25 points (the number of points used is a hyperparameter of the problem).

5 Acknowledgements

Credit to Olivier Shelbaya for conceptualization, many thanks to all the operators: Blair Stringer, Robert Shanks, Ed Parton, Chloe Beaumier-Martin, and Rene Tanaja for their feedback and support during MDEV. Thanks to the power supply and controls group including Dan Louie, Arthur CM. Leung, and Keiko Ezawa for their help diagnosing issues with DRA:Q1. Thanks to Spencer Kiy for support during MDEV.

References

[1] E. Ghelfi, A. Katrusiak, G. Kogler Anele, R. Baartman, W. Fedorko, O. Kester, O. Shelbaya and D. Tanyer. (2024). Bayesian Optimization for Ion Beam Centroid Correction. TRIUMF. Submitted for publication.

[2] S. Engel. (2003). Awakening of the DRAGON (PhD thesis). Ruhr-Universität Bochum. https://dragon.triumf.ca/docs/sabine_thesis.pdf

[3] D. Hutcheon, A. Olin, C. Ruiz. (2003). User's Manual for Tuning the DRAGON Separator. TRIUMF. https://dragon.triumf.ca/sep_tuning.html#3

[4] O. Shelbaya. (2023). Model Coupled Accelerator Tuning (PhD thesis). TRIUMF, UVic Dept. of Physics & Astronomy. https://dspace.library.uvic.ca/handle/1828/14804

6 Code

```
import json, time, requests, time
import numpy as np

session = requests.Session()
get_url = "https://vpn.beta.hla.triumf.ca/jaya/get"
```

```python
set_url = "https://vpn.beta.hla.triumf.ca/jaya/set"
check_url = "https://vpn.beta.hla.triumf.ca/jaya/checkAuth"

session.auth = requests.auth.HTTPBasicAuth(username, pw)

print(session.get(check_url,auth=(username,pw)).status_code)

wait_time = 0.5 # in seconds
current_vals = session.post(get_url, json={'readPvList': ['DRA:Q1:CUR', 'DRA:Q2:CUR']})
                            .json()['readPvDict']

setpoint_1 = float(current_vals['DRA:Q1:CUR'])
setpoint_2 = float(current_vals['DRA:Q2:CUR'])
# print(setpoint_1, setpoint_2)

set_pv = {'DRA:Q1:CUR': setpoint_1, 'DRA:Q2:CUR': setpoint_2}

# set_pv = {'DRA:Q1:CUR': setpoint_1+delta_1[indices[i%9][0]],
#            'DRA:Q2:CUR': setpoint_2+delta_2[indices[i%9][1]]}

payload = {'setPvDict': set_pv, 'waitTime': wait_time}
response = session.post(set_url, json=payload)
time.sleep(5)

# set combinations method
# delta_1 = [0.25*setpoint_1, 0, -0.25*setpoint_1]
# delta_2 = [0.25*setpoint_2, 0, -0.25*setpoint_2]
# indices = [[0,0], [0,1], [0,2], [1,0], [1,1], [1,2], [2,0], [2,1], [2,2]]

iterations = 1

# initial periods
p_1 = setpoint_1/20
p_2 = setpoint_2/20

if setpoint_1 >= 50 and setpoint_1 <= 300:
    a_1 = 50/setpoint_1

elif setpoint_1 > 300:
    a_1 = 350/setpoint_1 - 1

else:
    a_1 = 1

if setpoint_2 <= 50 and setpoint_2 >= 280:
    a_2 = 50/setpoint_2
```

```
elif setpoint_2 > 330:
    a_2 = 330/setpoint_2 - 1

else:
    a_2 = 1

i = 0

while(True):
    i = i + 0.5

    # set combinations method
    # set_pv = {'DRA:Q1:CUR': setpoint_1+delta_1[indices[i%9][0]],
    #           'DRA:Q2:CUR': setpoint_2+delta_2[indices[i%9][1]]}

    # sine method
    # delta_1 = np.sin(np.radians(200*i/setpoint_1))*setpoint_1*a_1
    # delta_2 = np.sin(np.radians(200*i/setpoint_2))*setpoint_2*a_2

    # triangle wave method
    delta_1 = a_1*setpoint_1*(1-2*abs(((i + p_1/4)%p_1)/(p_1/2)-1))
    delta_2 = a_2*setpoint_2*(1-2*abs(((i + p_2/4)%p_2)/(p_2/2)-1))

    set_pv = {'DRA:Q1:CUR': setpoint_1+delta_1, 'DRA:Q2:CUR': setpoint_2+delta_2}



    payload = {'setPvDict': set_pv, 'waitTime': wait_time}
    time.sleep(wait_time)
    response = session.post(set_url, json=payload)
    # print(response.json()['setPvDict'])
    print(float(response.json()['setPvDict']['DRA:Q1:CUR']))
```